XML Technology Planning Database: Lessons Learned

Raphael Some
Jet Propulsion Laboratory, California Institute of Technology
4800 Oak Grove Drive
Pasadena, CA 91109-8099
818-354-1902
Raphael.R.Some@jpl.nasa.gov
Jon Neff
The Aerospace Corporation
2350 E. El Segundo Blvd.
El Segundo, CA 90245
626-395-0465
Jon.M.Neff@aero.org

Abstract— A hierarchical XML database called XCALIBR (XML Capability Analysis LIBRary) has been developed by the New Millennium Program to assist in technology return on investment (ROI) analysis and technology portfolio optimization. The database contains mission requirements and technology capabilities, which are related by use of an XML dictionary. The XML dictionary codifies a standardized taxonomy for space missions, systems, subsystems and technologies. In addition to being used for ROI analysis, the database is being examined for use in project planning, tracking and documentation. During the past year, the database has moved from development into alpha testing. This paper describes the lessons learned during construction and testing of the prototype database and the motivation for moving from an XML taxonomy to a standard XML-based ontology.

TABLE OF CONTENTS

1. INTRODUCTION

NASA's New Millennium Program (NMP) [3] is chartered with the task of selecting high value, breakthrough technologies for future NASA science missions and maturing these technologies from the TRL 3-4 (breadboard) stage to TRL 7 (successful use in a flight system) [1,2]. In practice, the NMP technologists work with NASA Science Directorate technologists to define needed capabilities which can only be provided by advanced (i.e., beyond state of the art) technologies. These Technology Capability Areas (TCAs) are then used as the basis for open solicitations for technologies promising to provide these capabilities. The selection of technology providers is done through the NASA Research Announcement (NRA) process. The TCA identification and prioritization process, as well as the process used in evaluating individual technologies, has been, for the most part, a qualitative one, without a rigorous quantitative analysis by which relative rankings can be formulated, compared and understood.

To assist in the selection of high payoff TCAs and technologies, a means of providing a quantitative, traceable and defensible evaluation of expected benefit and return on investment (ROI) was desired. A team was formed to develop a methodology, database and tool set for performing these ROI evaluations.

The database, known as XCALIBR, is XML-based and hierarchical in nature. It contrasts sharply with traditional relational databases currently used to maintain these types of data in that it uses a tree structure, rather than a flat record file, to organize the data. The basis of the tree is a set of taxonomies, which describe the NASA organization, NASA space mission functions and structures, and a technology hierarchy. The taxonomies provide a hierarchical decomposition of each section of the database (Figure 3). Each node of the taxonomy tree contains a set of data defining that node's qualitative and quantitative descriptors or metrics. A single schema (data template) is used for all nodes, thus simplifying and unifying the database design (Figure 4). The data allowed in a node is defined by its node type (taxonomical identity). The node's taxonomical identity is used to personalize the generic schema for that specific node. The taxonomies, including descriptors, are embodied in an XML dictionary. The XML dictionaries rigorously define the data types allowed for each type of node and its associated descriptive metrics, and are the mechanism by which relationships between nodes such as NASA goals, mission requirements and technology capabilities are accessed, identified, and related. They define relationships between structural and functional entities and between organizations, missions and technologies. Thus, we use these XML dictionaries, and the taxonomies defined therein, to specify their conceptual relationships, or ontology. Using these XML tags allows, for instance, qualitative and quantitative comparison of technology capabilities to mission requirements.

The database allows population of arbitrary portions of mission requirements to different depths of the hierarchy. In this database, the users both enter data and determine the structure of the database, within the limits imposed by the taxonomy.

All fields in the database are machine-readable. Since XCALIBR serves as a common data repository for analysis tools, the database explicitly defines all types of data to allow these tools to access and unambiguously interpret the data.

In the following sections we discuss the current capabilities of the database and tool set as well as the lessons learned from alpha testing, user interface review, sample problem, and taxonomy development. We end with a discussion of future plans and conclusions.

2. CURRENT CAPABILITIES

Previous papers have covered the design of the database and the default analysis tool [4, 5]. This section covers the XCALIBR capabilities that are now operational. The section refers to Figure 1, starting from the bottom of the diagram and working up.

<u>Web Graphical User Interface</u>

The database has a Web-based user interface for navigating, querying, and entering data. (See user interface for mission requirements in Figure 5.) The user interface can be accessed through any standard Web browser and requires password authentication to log in. The user interface presents the user with hierarchically organized data. This contrasts sharply with the typically flat display of most relational databases. While the database provides the ability to define a hierarchical relationship between database elements, it is the user, who determines, at data entry time, the specific local hierarchical structure.

As shown in Figure 5, the display for each section of the database has navigation buttons across the top so that the user can go directly from one database area to another. Text below the top line of navigation buttons indicates which part of the database you are in. Below the text is an "address bar" that displays the exact location of the selected mission requirement in the hierarchy. The mission hierarchy is displayed as a tree in the left frame, while the detailed information for the selected requirement is displayed in the right frame.

The detailed requirement display has buttons at the top of the frame that allow the user to edit, add a child requirement, show a map of possible taxonomy paths from the selected requirement, add N levels of child requirements, delete the requirement, find technologies that fulfill the requirement, and launch the default analysis tool. A second row of buttons allows the user to add metrics, descriptors, and miscellaneous text notes.

The user interface for the technology side of the database is similar to the mission requirements interface. An interface for manually editing the taxonomy is also included.

<u>Tools Interface</u>

The system includes a standard interface for Excel-based tools. The interface initiates a set of recursive queries in the database that "walk down the tree" of mission requirements from a selected point in the hierarchy and return all technologies that match those requirements. The interface code then parses the results into a form that can be used by Excel and keeps track of the size and location of all data in the result set. Once the data is parsed, the interface populates an Excel template with the results, and custom code dynamically resizes the Excel spreadsheet to fit the analysis data.

The resulting Excel file is automatically downloaded to the user's machine. The raw input data and charts displaying the analysis results are included in the Excel file. The user then has an Excel-based model that he can modify to see how changes in requirement-technology matches affect return on investment. (See Figure 7.) Analysts can also substitute other evaluation algorithms to calculate technology value.

<u>Query Engine</u>

The XCALIBR system uses Qexo, an open-source Xquery processor, to process queries to the database [10]. The query engine uses predefined queries on data input by the user to find and return desired information.

<u>Taxonomy Dictionaries</u>

The heart of the XCALIBR system is a detailed taxonomy, including a data dictionary, which rigorously defines the structure and content of the database (Figures 2, 3). This taxonomy provides the means by which the database defines qualitative data, quantitative data and relationships. The database currently has a comprehensive taxonomy for spacecraft bus subsystems. The bus taxonomy covers everything from high system-level metrics to circuits and fasteners. Also included is a set of remote sensing and in-situ instruments and their associated metrics.

The entire taxonomy contains almost 700 elements, with each element representing a particular spacecraft subsystem, component, or part. Each element contains a unique set of metrics that characterize performance. Thus, the XCALIBR taxonomy has sufficient breadth to cover an entire space segment and sufficient depth to specify performance metrics at the component level and below.

NASA Organization, Mission, and Technology Data

The database includes a representation of the NASA organization to show how missions relate to directorates and their research goals (See Figure 5). XCALIBR is designed to be very flexible in response to both organizational and technical changes.

Both mission requirements and technologies are defined using the taxonomy. The use of the common taxonomy provides a means of relating lower-level technologies to higher-level capabilities, as well as automatically matching technologies to mission requirements.

In XCALIBR, specifying mission requirements consists of building the spacecraft representation using a straightforward top-down design approach familiar to system engineers. First, spacecraft requirements are specified. Then, subsystem requirements are specified based on the higher-level spacecraft requirements. The process is repeated for each subsystem component down to the desired level of detail. Of course, the user only needs to add data to elements which he wants to analyze.

The technology section of the database is organized by the taxonomy, with the taxonomy serving as a library "card catalog" system for filing technology data. As with mission data, the taxonomy also provides a standard set of performance metrics for each technology area.

By the time this paper is presented, the XCALIBR team will have populated the database with approximately ten sets of mission requirements and thirty technology capability areas to support NMP technology value analyses.

Tamino Database Engine

The XCALIBR system uses Tamino XML Server (TM) from Software AG. Tamino XML Server is a commercial database for storing, managing, publishing and exchanging XML documents in their native format, based on open-standard Internet technologies [11]. The XCALIBR Tamino database contains all of the taxonomy, mission, and technology data in the form of XML documents, along with the XML schemas that define the allowed structure for these documents. Access to the Tamino data store is provided by a custom Java application programming interface (API) developed by the XCALIBR team.

In summary, XCALIBR is now a useful working system that allows users to specify mission requirements and technology capabilities from the system level to the detailed component level.

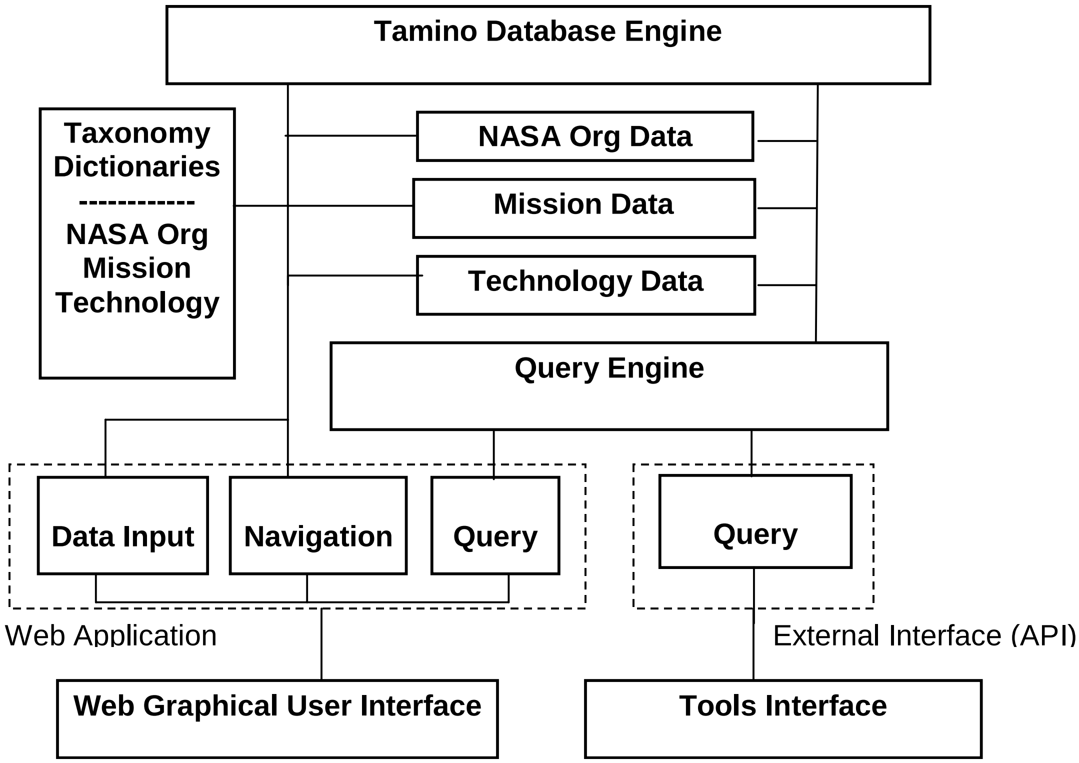

Figure 1 - Database Functional Block Diagram

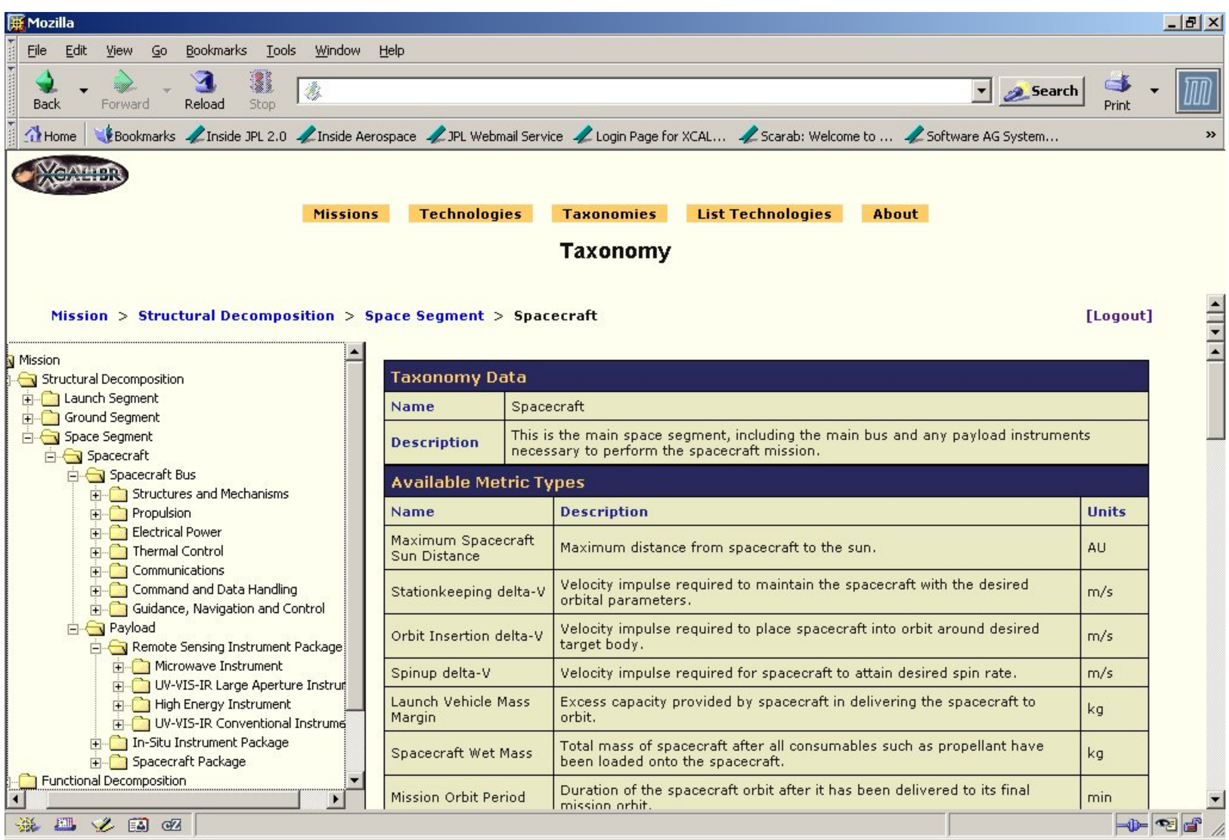

Figure 2 – User Interface for Browsing Taxonomy

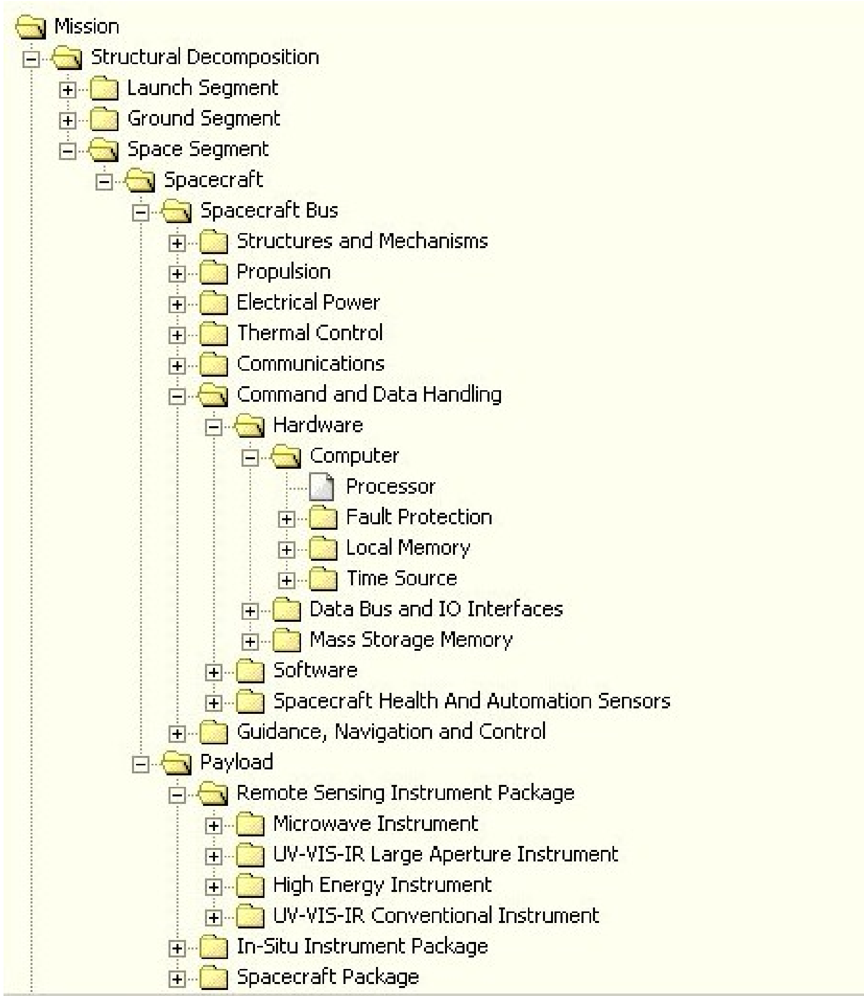

Figure 3- Partial Expansion of the XCALIBR Taxonomy

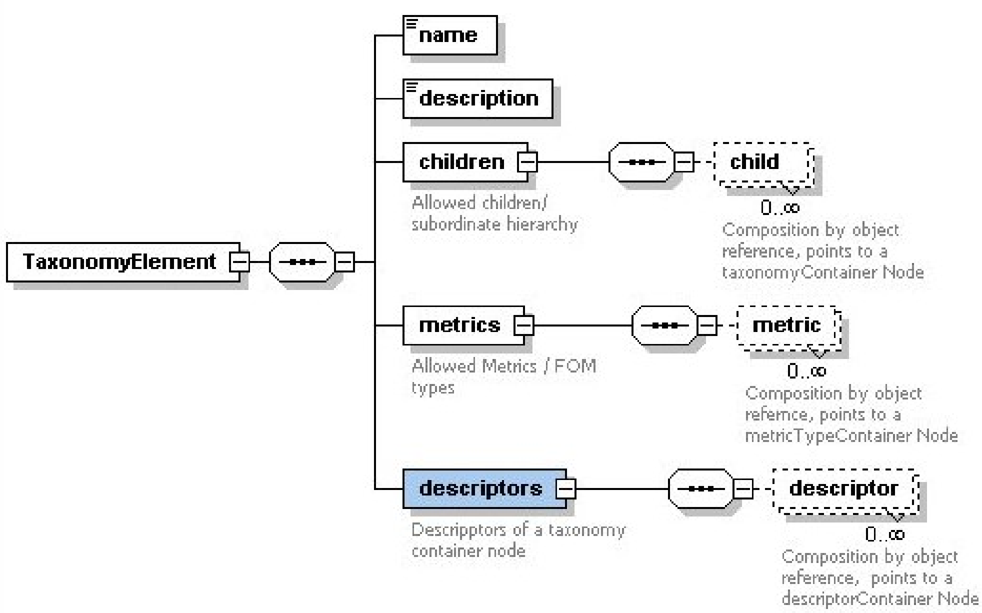

Figure 4- Taxonomy Element Schema

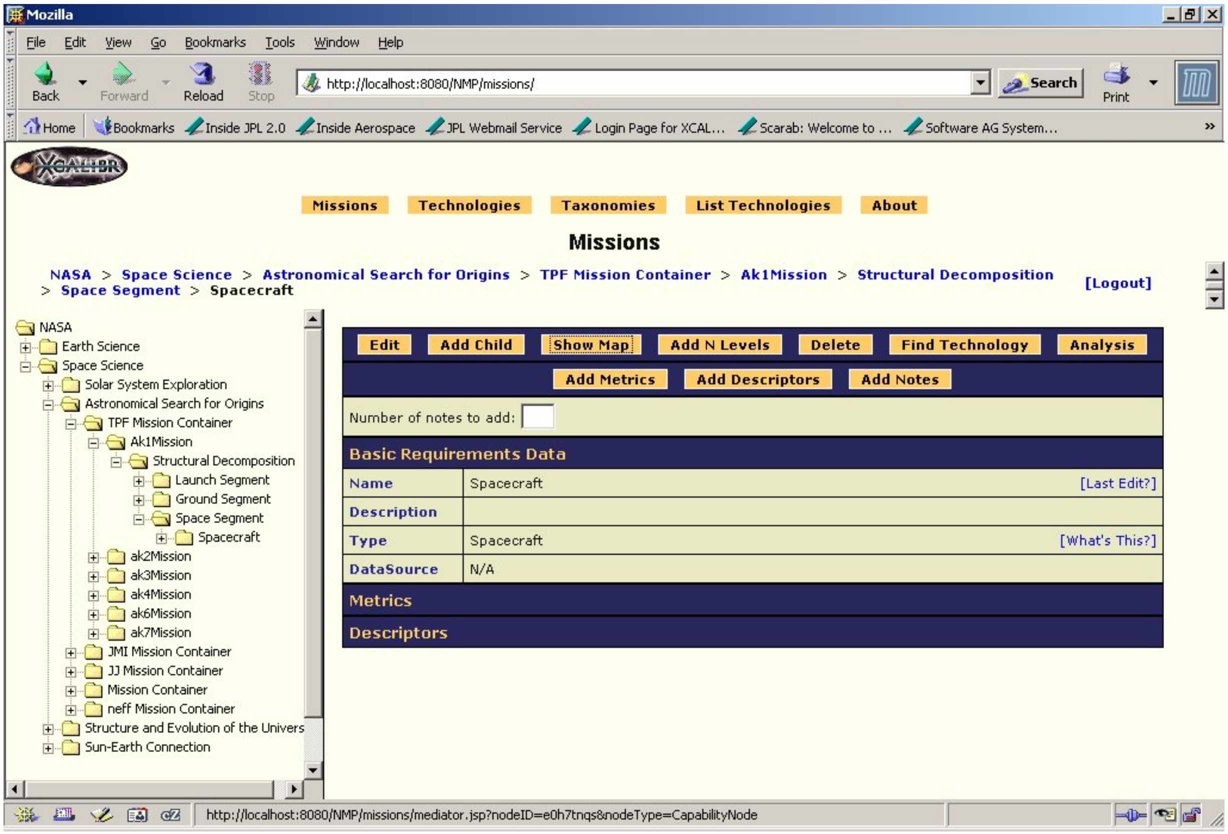

Figure 5- User Interface for Mission Requirements

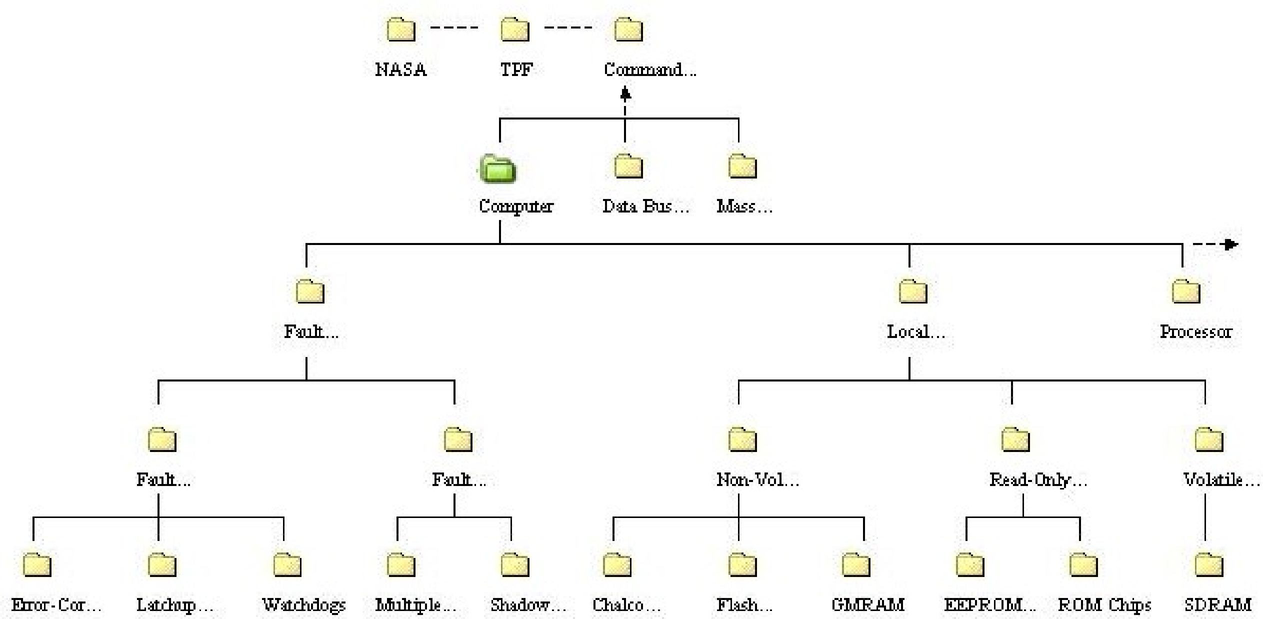

Figure 6- Proposed Block Diagram ("Org Chart") Navigation Interface

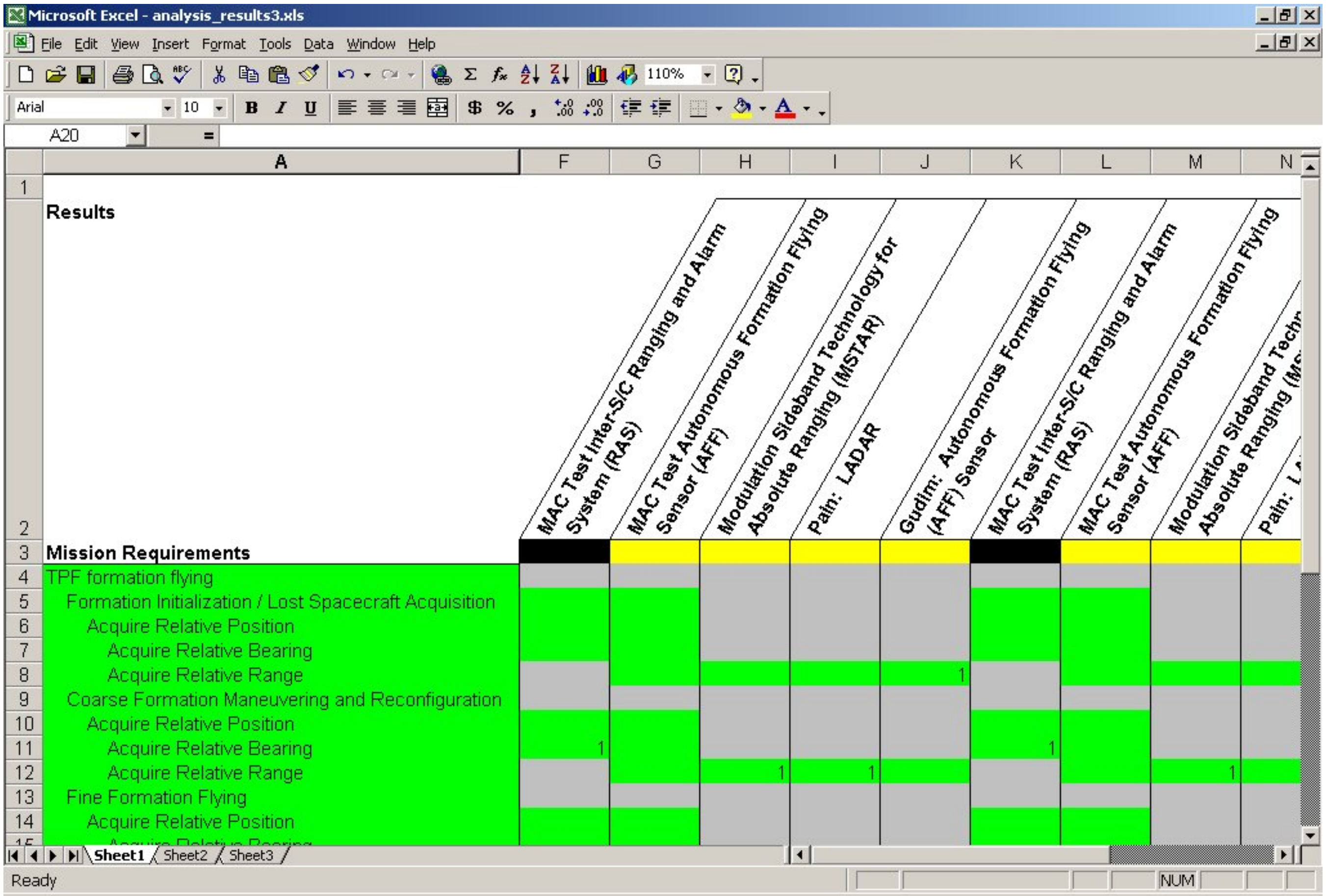

Figure 7- Analysis Tool: Input Data

3. ALPHA TESTING

In order to assess the usability of the XCALIBR system, alpha testing was performed with representatives from the target user community. After several months of development, the team produced a user interface using standard Web components that seemed simple and intuitive to the developers. However, one user's initial response was that he felt like he was playing Dungeons and Dragons (TM): exploring a maze with no idea what was in front of him and battling monsters at every turn. Thus the first lesson learned from alpha testing is that interfaces that seem obvious and intuitive to developers may be difficult and obscure for average users.

Navigation

Navigation was an issue that was identified early in the testing process. In order to use the database effectively, the user must be able to navigate through the data easily. A "folder tree" view is commonly used in many kinds of interfaces to display hierarchical information. (See Figures 2 and 5.) The alpha testers found this interface to be non-intuitive and difficult to use. They much preferred an "organization chart" or "block diagram" interface for navigation (Figure 6). The lesson is that the system needs to be flexible enough to support multiple views of the data and allow developers to add new views easily.

Data Entry

Once the users became familiar with the interface they began entering data. At this point one alpha tester encountered several issues related to the taxonomy. The user was immediately confronted with the dilemma of finding the correct place in the hierarchy to put the data. Most of the issues that the tester encountered while entering data were symptoms of a larger problem: how to fit vague, high-level mission requirements and technologies into a specific predefined taxonomy. The lesson learned here is that we need to do a better job of introducing users to the organizing principles of the taxonomy and provide better online hints for using the taxonomy.

There are some near-term solutions to this problem that were implemented quickly. The taxonomy was revised according to user inputs to improve organization and clarity. Certain areas of the taxonomy were expanded and more metrics were added, especially higher system-level metrics.

Also, a map function was added to show the user what was "around the corner of the maze." After analyzing one user's comments, it became apparent that choosing what kind of requirement to add was confusing. When a user adds a new mission requirement, he is presented with different options based on the taxonomy. However, being unfamiliar with the taxonomy, he did not know where the different options would lead, or what the implications of his choice were. The map function shows the region of the taxonomy that the user is operating in with the selected mission requirement node type as the root node. This new function should help orient the user to the taxonomy.

Some additional hands-on training and education was provided to the user to help show how the database was organized. Some of the problems perceived by the user went away after explanation. The original requirements for the XCALIBR system stated that untrained users should be able to do useful work five minutes after logging in for the first time. What we have learned is that, as with any advanced analysis tool, a user needs training in order to be effective.

Finally, during the additional training sessions the user realized that some of the mission requirements he was working with were not well-defined. We have learned that high-level mission requirements are frequently vague, especially for missions that are many years in the future. The lesson learned is that a taxonomy-based tool like XCALIBR forces mission designers to define terms more clearly.

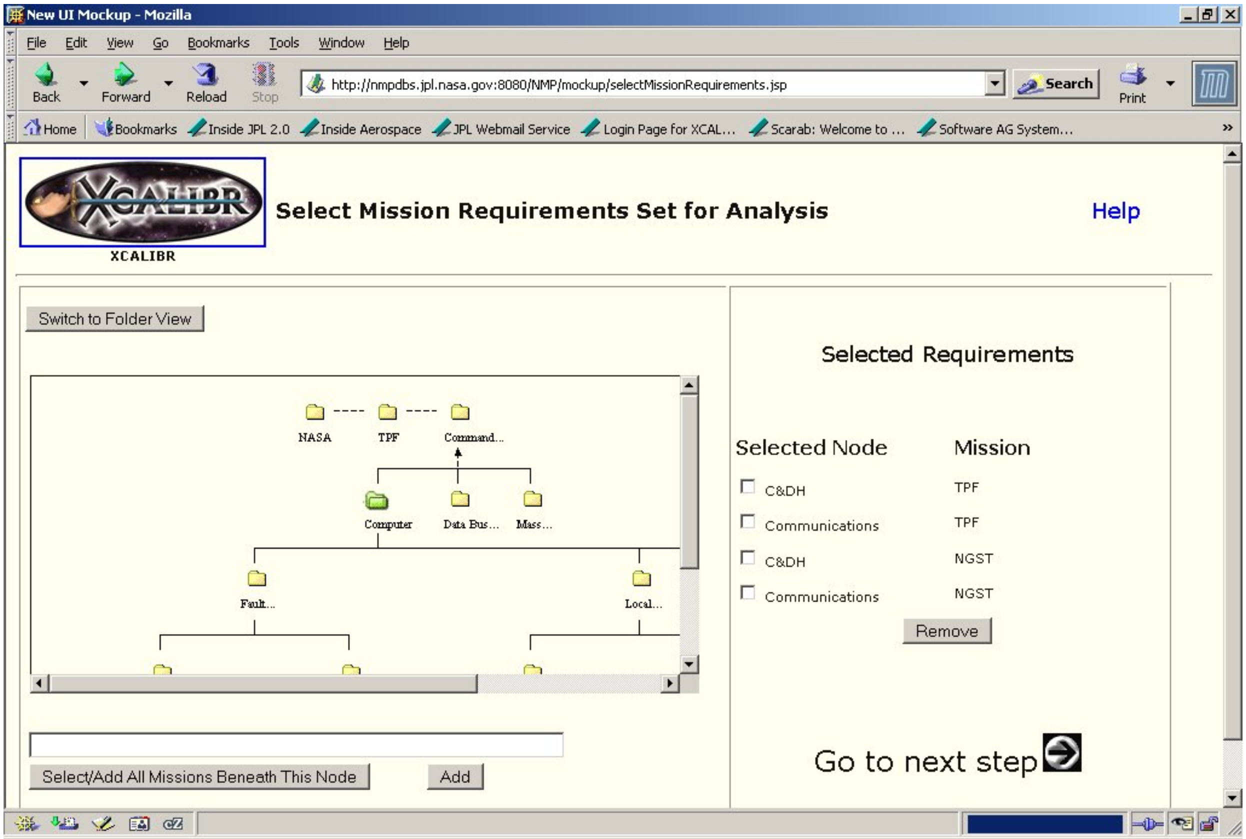

Figure 8 – Proposed Wizard for Selecting Mission Requirements for Analysis

4. USER INTERFACE REVIEW

In addition to alpha testing, the XCALIBR system was reviewed by a user interface consultant.

The reviewer found that the information architecture (taxonomy) was appropriate and effective, but made several recommendations for improving navigation. Interestingly, the reviewer recommended *not* using the org-chart style navigation interface (Figure 6) but rather keeping the folder tree. Note that system characteristics that the alpha tester found difficult the reviewer found intuitive, and vice-versa. Once again, the lesson is that the system needs to be flexible and support multiple views of the data. We currently plan to offer both folder tree navigation and org chart navigation interfaces and let each user decide what works best for him or her.

The majority of the recommendations involved the functional flow of the user interface. There were also a number of recommendations to improve consistency and clarity in labeling, formatting and terminology. Some terms that were clear to the team were confusing to the reviewer. Another lesson learned from the user interface review is choosing a clear set of terms is necessary, difficult, and requires user input.

5. SAMPLE PROBLEM

Two of the XCALIBR team members worked through a sample problem involving actual data used to conduct return-on-investment analyses for large space telescope technologies. The goals of the sample problem were to maximize the use of XCALIBR version 1.0 and the default analysis tool, use Excel as an adjunct for missing analysis capabilities, and evaluate system usability. The team reviewed four data sources and chose a previous study on large space telescopes. An example of the input data is shown in Figure 9.

<u>Problem Scope</u>

The scope of the sample problem was defined to include the following items:

- Determine the relative value of advances in seven technology capability areas for 10 meter, 35 meter, and 100 meter aperture missions. The technology capability areas included wavefront sensing and control, integrated modeling, robotic assembly, lightweight optics, thermal management, structures, and detectors.

- Explore performance value, technology readiness level (TRL), and cost as performance indicators. (This consisted of approximately 100 metrics spread across the 7 technology areas.)

Comparison of mission launch date with technology maturity was not performed and uncertainty (risk) was not considered, as these analyses were not part of the original study.

<u>Data Entry</u>

Data entry for the sample problem was performed manually using the Web interface and took about 80 hours. The data, including mission requirements and technologies, consisted of over 2000 discrete XML documents. Changes were made to the user interface to facilitate data entry, but it is clear that these changes alone will not significantly reduce the amount of time needed to enter large amounts of data. The lesson is that a graphical user interface is not sufficient for loading a large data set; more sophisticated data import tools are needed.

<u>Excel Analysis Tool</u>

The default analysis tool was not used because its capabilities did not match this problem. Instead, a custom Excel analysis tool was developed along with additional database queries to provide the required inputs to the evaluation algorithm. When the custom analysis tool was used on the input data, the results were virtually identical to the results obtained in the original study. The lesson is that the system must be able to handle multiple analysis tools.

<u>Taxonomy Issues</u>

On the surface, the sample problem inputs seemed to contain detailed data, such as cost, for each performance metric (Figure 9). This led to a rather large and unwieldy custom taxonomy to organize the space telescope data and a great deal of discussion about the appropriate level of detail in the database. Entering the new taxonomy branch for space telescopes into the database took about 40 hours and the new taxonomy elements were not consistent with the rest of the taxonomy. After closely examining the issue, the team discovered that the problem was actually the result of a difference in terminology between the authors of the original study and the XCALIBR team. However at this point the new taxonomy had already been developed and was not very useful for future studies.

The lesson is that users may need to re-formulate an analysis problem to fit the taxonomy structure. Simply putting together a custom taxonomy to solve a single problem violates a core purpose of XCALIBR, which is to encourage standardization and communication. Another significant lesson is that XCALIBR needs a well-defined process by which custom taxonomies can be normalized and migrated into the official taxonomy. The team is currently developing both software and documentation to improve this process.

6. Taxonomy and Ontology

The XCALIBR taxonomy is the heart of this information system. The taxonomy is the basis for database organization and constitutes the common language for defining requirements and technology capabilities. It is no surprise that the most interesting and difficult challenges in this task have been related to developing, displaying, and using the taxonomy.

<u>Taxonomy Browser</u>

Initially, the taxonomy was developed in either a word processing program or an XML editor and then converted into the XCALIBR database format using a semi-automated process. Eventually, tools were developed to export XML from the database, but the result was not easily readable by human beings. The lesson here is that a machine-readable format is usually not a good human-readable format. The team resolved this dilemma by creating a taxonomy browser as part of the Web user interface. The taxonomy browser provides the user with a human-readable interface to the machine-readable XML stored in the database.

<u>Taxonomy Editor</u>

Editing XML can be difficult, even with commercial editing tools. The XCALIBR taxonomy is not a monolithic document but rather a “virtual hierarchy” consisting of thousands of small XML documents linked by pointers. So, editing the XCALIBR taxonomy in native XML form is especially challenging. As new analysis problems were considered, the team learned that the taxonomy is a dynamic, living document. The taxonomy must constantly evolve to accommodate new types of data. This presents a special challenge because the taxonomy is also the foundation on which mission and technology data structures are built. Exporting the taxonomy, editing it, and then importing it back into the database takes about two weeks of manual editing and is prone to errors. The team addressed this issue by incorporating a taxonomy editor into the Web GUI. The taxonomy editor enables users to change the taxonomy without dealing with the underlying XML format.

Metrics

The current version of XCALIBR supports only scalar performance metrics. As long as metrics are scalars, they are fairly straightforward. A metric type has a name, a unique database index, descriptive text, and units of measurement. The team has learned, however, that many problems of interest to our users involve complex metric types such as probability distributions. The XCALIBR project is currently working on solutions to this issue.

Taxonomy Mapping

Taxonomies have existed for hundreds of years. Perhaps the most famous taxonomy is Carolus Linnaeus' classification system in biology. (In this system, all living things can be described according to kingdom, phylum, class, order, family, genus, and species.) With the advent of XML, taxonomies have become a cottage industry. The team has been asked repeatedly to compare the XCALIBR taxonomy to other NASA taxonomies. This is a never-ending task. The lesson learned is that the XCALIBR system must be able to easily map other taxonomies into its own internal taxonomy. The team is currently developing a standard set of processes and tools to manage the comparison and mapping of taxonomies.

Relationships Between Taxonomy Elements

The single biggest challenge in taxonomy development has been defining relationships between the various taxonomy elements. In a taxonomy tree structure, there is really only one relationship between elements, that of child to parent. However, in the current XCALIBR system the child to parent relationship can have several different meanings.

- Part to whole (the most common meaning)
- Specific type to generic type
- Mission to sponsoring organization
- Subsystem to system
- Lower level function to higher level function

The relationships between elements need to be defined clearly in order to avoid both conceptual problems and software problems. Distinguishing between "part of" and "type of" relationships is particularly important.

As shown by the recent set of NASA reorganizations, it is important to be able to respond rapidly to changes. (Missions and technologies are also constantly changing. Being able to handle technical changes is important also.)

Some relationships seem simple at a surface level but create a lot of confusion when implemented. An example is "symbolic links" that allow a single element to simultaneously be the child of multiple parents. For example, a single element representing a deployable mesh antenna might be part of both a communications subsystem and a microwave radiometer. What we have found is that this situation leads to ambiguity in determining the element's path, i.e., the element's position in the hierarchy. This ambiguity is problematic when running database queries, especially recursive queries. For this reason, we have abandoned the use of symbolic links for the time being.

As development proceeded, we realized that our users and customers needed to define relationships that could not be represented as a hierarchical tree. Rather, the desired rich relationship set formed a graph with multiple types of connections between elements.

The issues raised by different types of relationships between elements led the team to conclude that viewing the XCALIBR dictionary as a taxonomy tree was too limiting. The lesson we have learned over the past year is that the information architecture of this system needs to be based on *ontology* rather than taxonomy.

Both taxonomies and ontologies are structured hierarchical data dictionaries (called "vocabularies") that are used to make information more organized and easier to locate. Taxonomies grow out of library science and are aimed at helping *human beings* find and understand information. In the context of information architecture, ontologies originate in computer science and are designed to help *software* analyze, re-use, and exchange data. In an ontology, the fact that relationships are inherited makes it possible for software to infer relationships that are not explicitly stated, which is a very powerful feature. [12] An ontology "defines a common vocabulary for researchers who need to share information in a domain. It includes machine-interpretable definitions of basic concepts in the domain and the relations among them... An ontology together with a set of individual instances of classes constitutes a *knowledge base*." [7]

Moving from a taxonomy to an ontology offers many advantages for XCALIBR developers and users. Using a standard ontology language, such as OWL, makes the system more flexible because the schema does not need to be changed as new relationships or attributes are added. The object-oriented nature of modern ontology languages also makes it easier to add new technologies and hybrid devices to the database. A taxonomy helps the human user organize and understand information, but an ontology can be constructed to be both human-readable and machine-understandable. Because an ontology embeds an underlying conceptual model into the information architecture, it is possible to apply automated analysis and reasoning to reach conclusions that are not explicitly stated in the information itself.

							SOA			10m					
Area	Technology	weight	metric	unit		metric ref.	value	TRL	value ref.	value	STD	TRL	Cost ($M)	Years	value ref.
Structures		0.2											68.5		
	final structure	1											68.5		
			reflector area	m^2	positive	per definit	33	3	[3]	78.5	1	4	15	5	per defini
			stiffness (first mode)	Hz	positive	[2]	10	3	TRW kic	10	2	4	10	3	[2]
			damping	% of critic	positive	[2]	0.01	3	Material	0.1	0.01	4	5	3	[2]
			packing density	kg/m^3	positive	[1], p71	41	3	[3], calcu	40	3	4	10	5	EELV
			areal density (of S/C)	kg/m^2	negative	EELV limi	47.8	3	[12]	150	10	3	1	3	EELV
			RMS Error Correctable to	microns	negative	[10]	0.3	3	Marie Le	0.8	3	4	15	10	[10]
			Thermomechanical Stability	ppm	negative	NASA CR	5	1	[10]	3	1	3	5	5	[10]
			Microdynamic Stability and H	ppm	negative	[11]	5	1	[11]	3	3	4	7.5	7.5	[11]

Figure 9 – Example of Input Data for Space Telescope Sample Problem (Structures Technology Area)

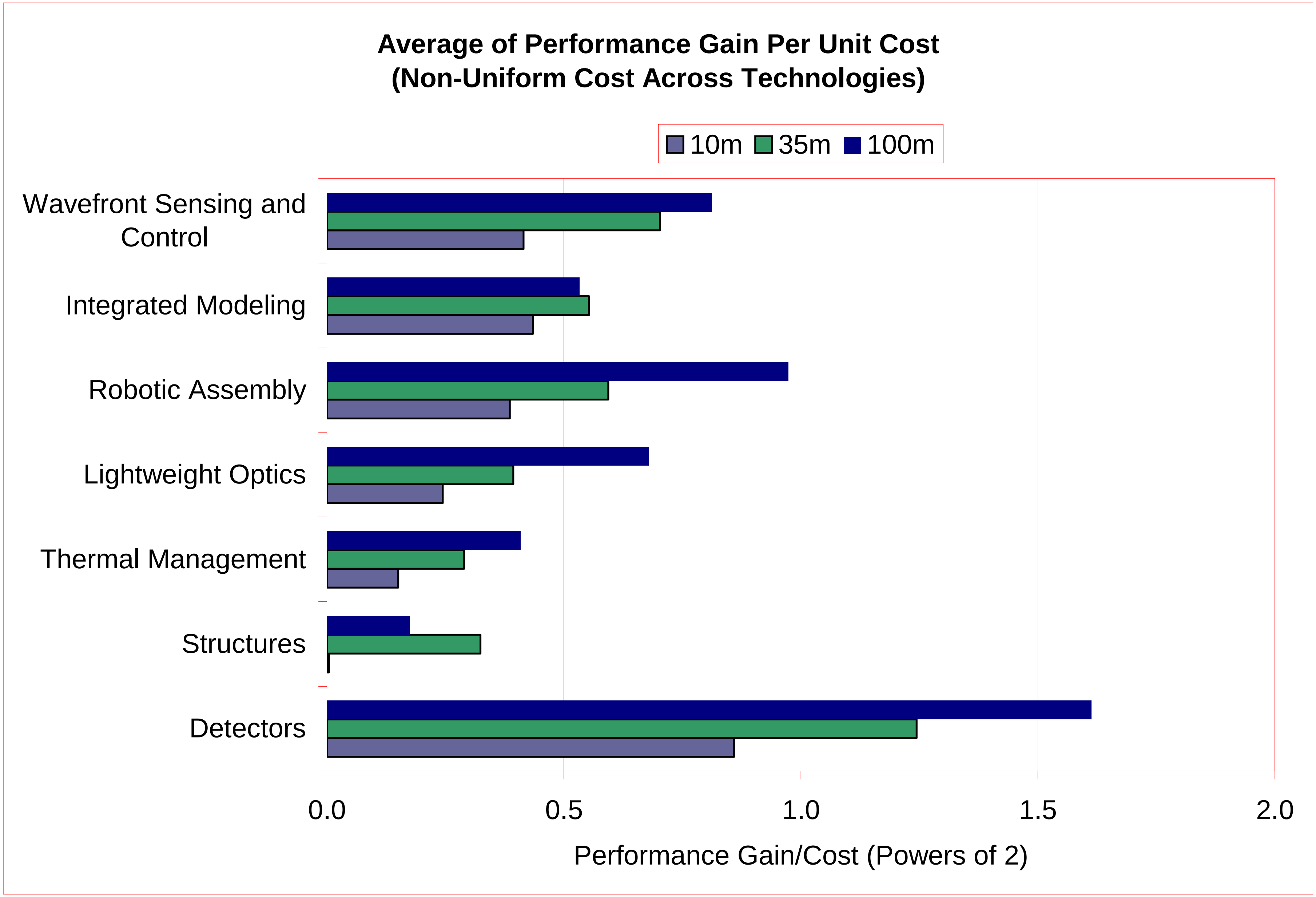

Figure 10- Sample Problem Results

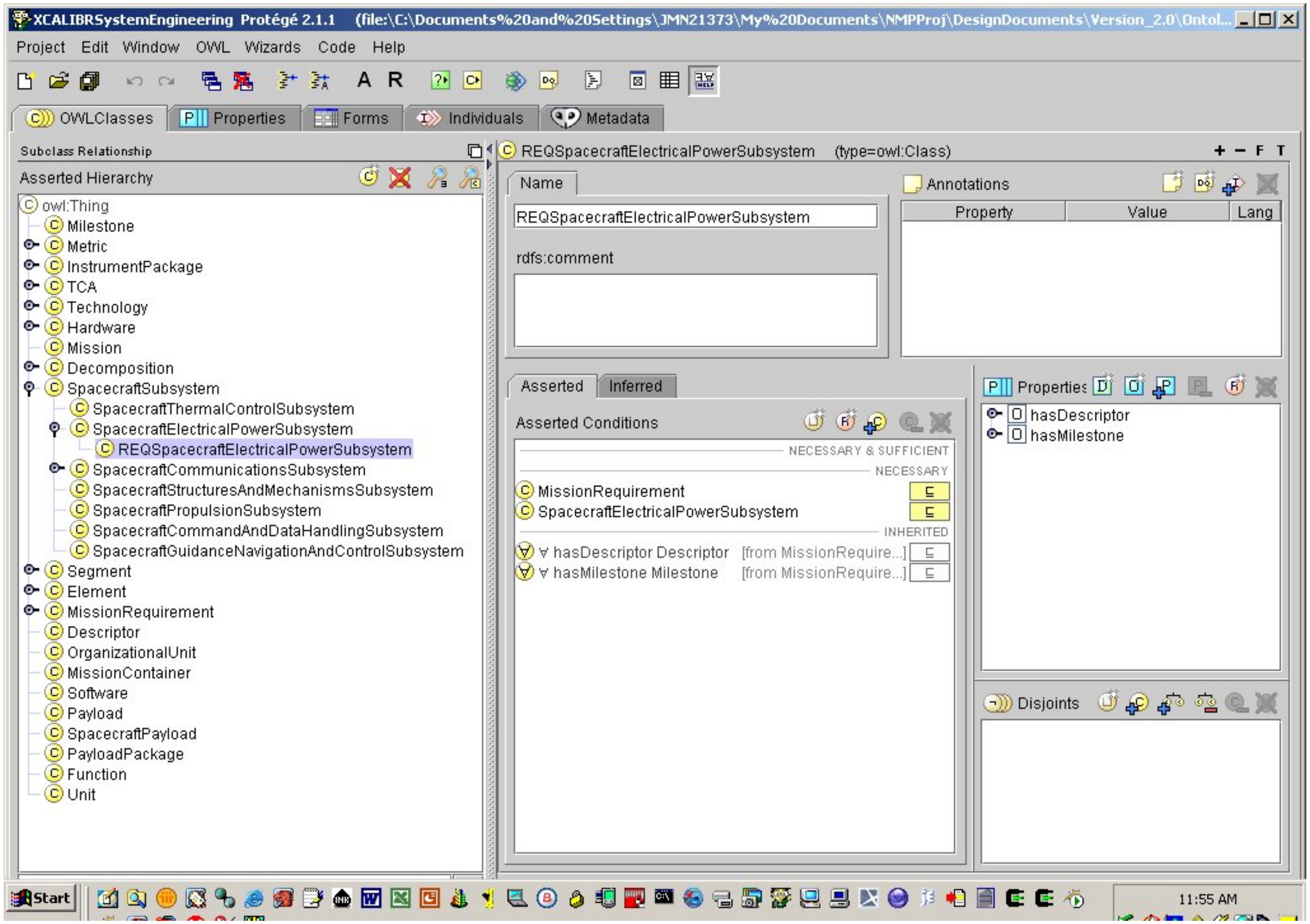

Figure 11- Prototype XCALIBR Ontology in Protege

0-7803-8155-6/04/$17.00©2005 IEEE

7. FUTURE PLANS

Version 1.x

Our current plan is to use version 1.x as a development testbed to try out new ideas for the user interface, such as “org chart” style navigation. Version 1.x will also be used operationally for the ST-10 evaluation. (ST-10 is the next NMP mission, currently in pre-formulation phase.) The ST-10 evaluation will provide a beta test for the system.

The ST-10 analysis is the first time that XCALIBR is being used on an actual evaluation task. The team has already mapped related taxonomies into the XCALIBR taxonomy and is in the process of collecting mission requirements. The evaluation will proceed in a series of approximately 2-month iterations, with additional areas being included in the analysis at each iteration. The first iteration focuses on mission requirements and technologies for telecommunications, extreme environments, and planetary surface operations such as mobility and sample collection. In the process of collecting, organizing, and entering data, the team will refine and standardize new taxonomy areas used for this analysis. Future iterations will add additional technology areas for analysis. The ST-10 evaluation is expected to continue through the end of FY05.

Version 2.x

Requirements and high-level architecture design are complete for version 2.0 and the team has begun development. The new XCALIBR architecture will leverage existing components and technologies such as Protege (Figure 11), WebDAV, OWL, and Java Server Faces (JSF) [7, 8, 9, 10]. Several major features, which were not feasible under the old 1.x architecture, will be implemented in 2.0. The new version will have

- An ontology-based information architecture that will resolve the difficult relationship issues encountered while developing the 1.x taxonomy.
- An improved user interface, which includes wizards to aid the user in completing data entry and analysis tasks, as well as online help.
- A complete security model, including restriction of administrator-defined database areas to a set of authorized users.
- A concurrency model, which will prevent users from editing the same database entries at the same time.
- An undo feature, which will allow users to roll back changes to the database.
- A history mechanism, which will preserve all changes to the database so that users can see how requirements and technology capabilities have evolved over time.
- An import/export tool to facilitate moving large amounts of data into and out of the database.

Other Applications

In the future, XCALIBR could be adapted to serve purposes other than technology benefit analysis. For instance, version 2.0 could also be used as a tool for documenting mission requirements and design from the system level down to the detailed component level. XCALIBR could also document the as-built design in the same system, allowing system engineers and technologist to more easily draw lessons from the evolution of the system design. XCALIBR 2.0 will have the ability to track budgets and schedules, which could lead to integrated technical and programmatic analysis. The system could also be used to track project technology selections.

All of these functions could be included in a single tool that

- is accessible from any Web browser
- is intuitive and easy to use
- provides a rigorous, well-defined common language across mission designers, project managers, technologists
- provides an index to a repository of project documents

8. CONCLUSIONS

A new kind of database has been developed to conduct technology benefit analyses for the New Millennium Program. The foundation of this database is an XML taxonomy that has broad application across the aerospace sector. The database and user interface have been tested with potential users, reviewed by software experts, and exercised with sample problems.

The database system has completed alpha testing with a subset of the target users. XCALIBR has been used to solve a sample technology evaluation problem and the results have been verified by the results of a previous analysis that did not use XCALIBR. The system has also undergone a user interface review by a professional GUI designer.

The database is currently being used operationally to help evaluate technology areas to be considered for flight-testing on the ST-10 mission.

A new ontology-based system is being developed for FY05. The ontology being developed for this new system, if accepted and ratified, will, for the first time, provide a common language, allowing free and unambiguous exchange of information, across all sectors of the aerospace community.

9. ACKNOWLEDGEMENTS

This work was carried out at the Jet Propulsion Laboratory, California Institute of Technology, under a contract with the National Aeronautics and Space Administration.

The authors would like to acknowledge the help of the XCALIBR team in providing inputs to this paper. In particular, we wish to thank Mark Cowdin, Jean Hester and Phil Moynihan for their insights and observations.

REFERENCES

[1] R. Mackey, R. Some, A. Aljabri, "Readiness Levels for Spacecraft Information Technologies" *Proceedings of the 2003 IEEE Aerospace Conference*, March 2003.

[2] C. Minning, P Moynihan, J. Stocky, "Technology Levels For The New Millennium Program", *Proceedings of the 2003 IEEE Aerospace Conference*, March 2003.

[3] C. Minning, D. Crisp "NASA's New Millennium Program: Flight Validation of Advanced Technologies for Space Applications", *Journal Of Space Mission Architecture, Vol. 3*, Fall 2003.

[4] A. Czikmantory, M. Marshall, J. Neff, R. Some, "XML Hierarchical Database for Missions and Technologies ", *Proceedings of the 2004 IEEE Aerospace Conference*, March 2004.

[5] J. Neff, R. Some, C. Weisbin, J. Witz, "XML Based Tools for Assessing Potential Impact of Advanced Technology Space Validation ", *Proceedings of the 2004 IEEE Aerospace Conference*, March 2004.

[6] OWL Web Ontology Language Overview W3C Recommmendation, http://www.w3.org/TR/owl-features/.

[7] D. McGuinness, N. Noy, "Ontology Development 101: A Guide to Developing Your First Ontology", http://protege.stanford.edu/publications/ontology_development/ontology101.html.

[8] Protege Web site, http://protege.stanford.edu/.

[9] WebDAV Resources Web Site, http://www.webdav.org/.

[10] Qexo - The GNU Kawa Implementation of Xquery Web Site, http://www.gnu.org/software/qexo/.

[11] M. Champion and F. Jung, "Tamino XML Server," Software AG White Paper, August 2003. http://www.softwareag.com/tamino/download/e-WP_XMLserver_INS-WP17E0803.pdf

[12] K. Adams, "The Semantic Web: Differentiating Between Taxonomies and Ontologies," Online, July/August 2002. http://www.iaslash.org/node.php?id=4592.

BIOGRAPHY

Raphael Some *has over 27 years experience in technology planning, development and commercialization. He is a Level 1 Manager at the Jet Propulsion Laboratory, where he currently serves as avionics and microelectronics technologist in NASA's New Millennium Program. Prior positions at JPL include Chief Engineer and Systems Engineering Manager of the REE project and initiator of JPL's Smart Sensor Web. Prior to joining JPL, Raphael held a variety of positions in the aerospace industry including VP of Engineering at 3D Micro, Chief Engineer and Technical Director at Irvine Sensors Corporation., and Technical Director at Raytheon Corporation.*

Jon M. Neff *is a Senior Project Engineer in the Civil and Commercial Division of The Aerospace Corporation. Previously he worked at the Jet Propulsion Laboratory and as an engineering manager at small software companies. His areas of interest include software engineering, strategic planning for technology development, and remote sensing. Jon has taught risk management and decision theory in The Aerospace Institute's System Architecture/System Engineering Certificate Program. Jon has received several awards from NASA and The Aerospace Corporation for his work in software engineering, technology planning, and project management. He has BS, MS, and PhD degrees in*

Aerospace Engineering from the University of Texas at Austin, and an MBA from Pepperdine University.

0-7803-8155-6/04/$17.00©2005 IEEE